**Отчет по лабораторной работе № 8-9** по курсу Фундаментальная информатика

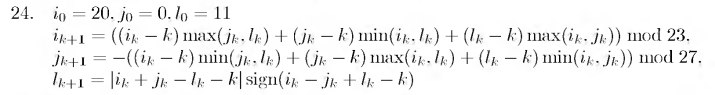
Студент группы: **М8О-101Б-22, Чапалда Мария Олеговна**, № по списку: **24**, Контакты **mariyaaach03@gmail.com** Работа выполнена: « » \_\_\_\_\_\_\_\_\_\_\_\_202\_\_\_г.

Преподаватель: **каф. 806 Крылов Сергей Сергеевич**, Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Отчет сдан « » \_\_\_\_\_\_\_\_\_202 \_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 **Тема:**  Программирование на языке Си

1. **Цель работы:** составить и отладить простейшую программу на СИ. Изучить возможности компилятора
2. **Задание** (*вариант №* **24**)**:** 

4 **Оборудование**

**Оборудование *ПЭВМ студента, если использовалось****:* **16 Гб оперативной памяти, экран диагональю 17.3", с разрешением экрана 1920x1080 Пикс**

Процессор  **AMD Ryzen 7 3750H with Radeon Graphics 3.20 GHz**

*Оборудование ПЭВМ студента, если использовалось:*

Процессор \_AMD Ryzen 7 2700\_ с ОП 16 Гб НМД \_\_\_5\_\_ Тб. Монитор 1920x1080~60Hz

Другие устройства \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**6. Идея, метод, алгоритм** решения задачи(в формах:словесной,псевдокода,графической[блок-схема,диаграмма,рисунок,таблица] или формальные спецификации с пред- и постусловиями)

Алгоритм:

1. подключение библиотек вводы и вывода математических функций
2. Задаём константные переменные, а именно координаты вершин треугольника
3. функция максимума из 2 чисел
4. функция которая определяет знак числа
5. функция модуля
6. функция минимума из 2 чисел
7. Ввожу счетчик точек, которые попадут в треугольник после выполнения программы
8. Создаю три массива
9. Присваиваю первому элементу массива значение i,j,l первых
10. В цикле вычисляю элемента под индексом k+1 для i,j,l
11. Применяю векторное и псевдоскалярное произведение, которому свойственно следующее: если они одинакового знака, то точка внутри треугольника, если что-то из этого - ноль, то точка лежит на стороне, иначе точка вне треугольника

**7. Сценарий выполнения работы** [план работы,первоначальный текст программы в черновике(можно на отдельном листе)итесты либо соображения по тестированию].

1. Составил алгоритм, описан в пункте 6
2. Реализовать его в программе
3. Провести тесты

*Пункты 1-7 отчета составляются строго до начала лабораторной работы*

**8. Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами,подписанныйпреподавателем).

Код программы:

//подключение библиотек вводы и вывода математических функций

#include <stdio.h>

#include <math.h>

//задаём константные переменные, а именно координаты вершин треугольника

#define X1 -10

#define Y1 0

#define X2 0

#define Y2 10

#define X3 -10

#define Y3 20

//функция максимума из 2 чисел

int max(int a, int b)

{

    return a > b ? a : b;

}

//функция которая определяет знак числа

int sign(int a)

{

    int result = 0;

    if (a > 0) {

        result = 1;

    }

    else if (a < 0) {

        result = -1;

    }

    return result;

}

//функция модуля

int abs(int a)

{

    return a >= 0 ? a : -a;

}

//функция минимума из 2 чисел

int min(int a, int b)

{

    return a < b ? a : b;

}

//основная функции

int main()

{

    //вводим счетчик точек, которые попадут в треугольник после выполнения программы

    int count = 0;

    //создаём три массива

    int i[50];

    int j[50];

    int l[50];

    //присваиваем первому элементу массива значение i,j,l первых

    i[0] = 20;

    j[0] = 0;

    l[0] = 11;

    //цикл

    for (int k = 0; k < 50; ++k) {

        //вычисление элемента под индексом k+1 для i, j, l

        i[k + 1] = ((i[k] - k) \* max(j[k], l[k]) + (j[k] - k) \* min(i[k], l[k]) + (l[k] - k) \* max(i[k], l[k])) % 23;

        j[k + 1] = -((i[k] - k) \* min(j[k], l[k]) + (j[k] - k) \* max(i[k], l[k]) + (l[k] - k) \* min(i[k], l[k])) % 27;

        l[k + 1] = abs(i[k] + j[k] - l[k] - k) \* sign(i[k] - j[k] + l[k] - k);

        //применяем векторное и псевдоскалярное произведение, которому свойственно следующее: Если они одинакового знака, то точка внутри треугольника, если что-то из этого - ноль, то точка лежит на стороне, иначе точка вне треугольника

        if ((((X1 - i[k]) \* (Y2 - Y1) - (X2 - X1) \* (Y1 - j[k]) >= 0) &&

           ((X2 - i[k]) \* (Y3 - Y2) - (X3 - X2) \* (Y2 - j[k]) >= 0) &&

           ((X3 - i[k]) \* (Y1 - Y3) - (X1 - X3) \* (Y3 - j[k]) >= 0 )) || (((X1 - i[k] )\* (Y2 -Y1 ) - (X2 - X1) \* (Y1 - j[k]) < 0) &&

           ((X2 - i[k]) \* (Y3 - Y2) - (X3 - X2) \* (Y2 - j[k]) < 0) && ((X3 - i[k]) \* (Y1 - Y3) - (X1 - X3) \* (Y3 - j[k]) < 0))) {

            count += 1;

            printf("nomer tochki=%d\n", count);

            printf("x=%d\n", i[k]);

            printf("y=%d\n", j[k]);

            printf("diskretnoe vremya=%d\n", k);

            break;

        }

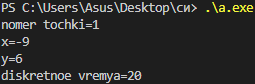
    }

    //пишем return, так как функция типа "int" должна что-либо возвращать

    return 0;

}

Запуск программы:



**9. Дневник отладки** должен содержать дату и время сеансов отладки и основные события(ошибки в сценарии и программе,нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Замечания автора** по существу работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Выводы**

\_\_\_\_В ходе выполнения этой лабораторной работы я научилась составлять простейшие программы на языке Си

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_